$ AF/2136
FW

HEWLETT-PACKARD COMPANY
Intellectual Property Administration
P.O. Box 272400
Fort Collins, Colorado 80527-2400

PATENT APPLICATION

ATTORNEY DOCKET NO. 10019968-1

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Inventor(s):	Neal A. KRAWETZ	Confirmation No.:	9182
Application No.:	09/975,815	Examiner:	Colin, Carl G.
Filing Date:	October 11, 2001	Group Art Unit:	2136

Title: SYSTEM AND METHOD FOR SECURE DATA TRANSMISSION

Mail Stop Appeal Brief-Patents
Commissioner For Patents
PO Box 1450
Alexandria, VA 22313-1450

TRANSMITTAL OF APPEAL BRIEF

Transmitted herewith is the Appeal Brief in this application with respect to the Notice of Appeal filed on February 27, 2006.

The fee for filing this Appeal Brief is (37 CFR 1.17(c)) $500.00.

(complete (a) or (b) as applicable)

The proceedings herein are for a patent application and the provisions of 37 CFR 1.136(a) apply.

□ (a) Applicant petitions for an extension of time under 37 CFR 1.136 (fees: 37 CFR 1.17(a)-(d)) for the total number of months checked below:

□ 1st Month $120 □ 2nd Month $450 □ 3rd Month $1020 □ 4th Month $1590

□ The extension fee has already been filed in this application.

☒ (b) Applicant believes that no extension of time is required. However, this conditional petition is being made to provide for the possibility that applicant has inadvertently overlooked the need for a petition and fee for extension of time.

Please charge to Deposit Account 08-2025 the sum of $ 500. At any time during the pendency of this application, please charge any fees required or credit any over payment to Deposit Account 08-2025 pursuant to 37 CFR 1.25. Additionally please charge any fees to Deposit Account 08-2025 under 37 CFR 1.16 through 1.21 inclusive, and any other sections in Title 37 of the Code of Federal Regulations that may regulate fees. A duplicate copy of this sheet is enclosed.

☒ I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to: Commissioner for Patents, Alexandria, VA 22313-1450

Date of Deposit: April 27, 2006

OR

□ I hereby certify that this paper is being transmitted to the Patent and Trademark Office facsimile number (571)273-8300.

Date of facsimile:

Typed Name: Joan B. Farragher

Signature: J Farragher

Respectfully submitted,

By James L Baudino
James L. Baudino
Attorney/Agent for Applicant(s)

Reg No. : 43,486

Date : April 27, 2006

Telephone : (214) 855-7544

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

APPEAL FROM THE EXAMINER TO THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re Application of:	Neal A. KRAWETZ	Confirmation No.:	9182
Serial No.:	09/975,815		
Filing Date:	October 11, 2001		
Group Art Unit:	2136		
Examiner:	Colin, Carl G.		
Title:	SYSTEM AND METHOD FOR SECURE DATA TRANSMISSION		
Docket No.:	10019968-1		

MAIL STOP: APPEAL BRIEF PATENTS
Commissioner for Patents
P.O. Box 1450
Alexandria, Virginia 22313-1450

Dear Sir:

APPEAL BRIEF

Applicant has appealed to the Board of Patent Appeals and Interferences from the decision of the Examiner mailed November 30, 2005, finally rejecting Claims 1-34. Applicant filed a Notice of Appeal on February 27, 2006. Applicant respectfully submits herewith this Appeal Brief with authorization to charge the statutory fee of $500.00.

REAL PARTY IN INTEREST

The present application was assigned to Hewlett-Packard Company as indicated by an assignment from the inventor recorded on March 14, 2002 in the Assignment Records of the United States Patent and Trademark Office at Reel 012730, Frame 0935. The present application was subsequently assigned to Hewlett-Packard Development Company, L.P. as indicated by an assignment from Hewlett-Packard Company recorded on September 30, 2003 in the Assignment Records of the United States Patent and Trademark Office at Reel 014061, Frame 0492. The real party in interest is Hewlett-Packard Development Company, LP, a limited partnership established under the laws of the State of Texas and having a principal place of business at 20555 S.H. 249 Houston, TX 77070, U.S.A. (hereinafter "HPDC"). HPDC is a Texas limited partnership and is a wholly-owned affiliate of Hewlett-Packard Company, a Delaware Corporation, headquartered in Palo Alto, CA. The general or managing partner of HPDC is HPQ Holdings, LLC.

RELATED APPEALS AND INTERFERENCES

There are no known appeals or interferences that will directly affect or be directly affected by or have a bearing on the Board's decision in this pending appeal.

STATUS OF CLAIMS

Claims 1-34 stand rejected, pursuant to a Final Office Action mailed November 30, 2005. Claims 1-34 are presented for appeal.

STATUS OF AMENDMENTS

No amendment has been filed subsequent to the mailing of the Final Office Action.

SUMMARY OF CLAIMED SUBJECT MATTER

Embodiments of the present invention as defined by independent Claim 1 are directed toward a method for secure data (58, 104) transmission comprising: generating a character string (54, 116) at a sender (14, 16, 18, 20); generating a hash key (64, 118) using the character string (54, 116) and a private key (62, 110); encrypting the data (58, 70, 72, 104, 112, 114) using the hash key (64, 118); and transmitting an identification key (60, 108) associated with the sender (14, 16, 18, 20), the character string (54, 116), and the encrypted data (72, 112) from the sender (14, 16, 18, 20) to a recipient (14, 16, 18, 20). (at least at page 3, lines 11-31; page 4, lines 3-30; page 5, lines 4-32; page 6, lines 1-16 and lines 25-32; page 7, lines 1-4, lines 13-22 and lines 27-31; page 8, lines 1-11; and figures 1-3).

Embodiments of the present invention as defined by independent Claim 11 are directed toward a method for secure data (58, 70, 72, 104) transmission comprising receiving a character string (54, 116) from a sender (14, 18), receiving an identification key (60, 108) from the sender (14, 18), receiving encrypted data (72, 112) from the sender (14, 18), determining a private key (110) associated with the sender (14, 18) using the identification key (60, 108), and decrypting the encrypted data (72, 112) using the private key (110) and the character string (54, 116). (at least at page 3, lines 11-31; page 4, lines 3-30; page 5, lines 4-32; page 6, lines 1-16 and lines 25-32; page 7, lines 1-4, lines 13-22 and lines 27-31; page 8, lines 1-11; and figures 1-3).

Embodiments of the present invention as defined by independent Claim 19 are directed toward a system (10) for secure data (58, 70, 72) transmission comprising: a processor (30); a memory (32) coupled to the processor (30); a string generator (40) stored in the memory (32) and executable by the processor (30), the string generator (40) adapted to generate a character string (54); a hashing engine (42) stored in the memory (32) and executable by the processor (30), the hashing engine (42) adapted to generate a hash key (64) using the character string (54) and a private key (62); and an encryption engine (44) stored in the memory (32) and executable by the processor (30), the encryption engine (44) adapted to encrypt the data (58, 70, 72) using the hash key (64); and wherein the processor (30) is adapted to transmit the encrypted data (72), an identification key (60) related to the private key (62), and the character string (54) to a recipient (16, 20). (at least at page 3, lines 11-31; page 4, lines 3-30; page 5, lines 4-32; page 6, lines 1-16 and lines 25-32; page 7, lines 1-4, lines 13-22 and lines 27-31; page 8, lines 1-11; and figures 1-3).

Embodiments of the present invention as defined by independent Claim 27 are directed toward a system (10) for secure data (58, 70, 72, 104) transmission comprising: a processor (80) adapted to receive encrypted data (72, 112), an identification key (60, 108), and a character string (54) from a sender (14, 18); a memory (82) coupled to the processor (80); a relational database (100, 102) stored in the memory (82) and accessible by the processor (80), the relational database (100, 102) relating the identification key (60, 108) to a private key (62, 110); and a decryption engine (88) stored in the memory (82) and executable by the processor (80), the decryption engine (88) adapted to decrypt the encrypted data (72, 112) using the character string (54, 116) and the private key (62, 110). (at least at page 3, lines 11-31; page 4, lines 3-30; page 5, lines 4-32; page 6, lines 1-16 and lines 25-32; page 7, lines 1-4, lines 13-22 and lines 27-31; page 8, lines 1-11; and figures 1-3).

GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

1. Claims 1-34 are rejected under 35 U.S.C. §103(a) as being unpatentable over U.S. Patent No. 5,757,915 issued to Aucsmith et al. (hereinafter "*Aucsmith*") in view of U.S. Patent Publication No. 2002/0094085 to Roberts (hereinafter "*Roberts*").

ARGUMENT

A. Standard

1. 35 U.S.C. § 103

To establish a *prima facie* case of obviousness under 35 U.S.C. § 103, three basic criteria must be met: First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings; second, there must be a reasonable expectation of success; and finally, the prior art reference (or references when combined) must teach or suggest all the claim limitations. *In re Vaeck*, 947 F.2d 488, (Fed. Cir. 1991); M.P.E.P. § 2143. The teaching or suggestion to make the claimed combination and the reasonable expectation of success must both be found in the prior art, and not based on applicant's disclosure. *Id.* Further, the mere fact that references can be combined or modified does not render the resultant combination obvious unless the prior art also suggests the desirability of the combination. *In re Mills*, 916 F.2d 680 (Fed. Cir. 1990); M.P.E.P. § 2143.01. Additionally, not only must there be a suggestion to combine the functional or operational aspects of the combined references, but also the prior art is required to suggest both the combination of elements and the structure resulting from the combination. *Stiftung v. Renishaw PLC*, 945 F.2d 1173, 1183 (Fed. Cir. 1991). Moreover, where there is no apparent disadvantage present in a particular prior art reference, then generally there can be no motivation to combine the teaching of another reference with the particular prior art reference. *Winner Int'l Royalty Corp. v. Wang*, 202 F.3d 1340, 1349 (Fed. Cir. 2000).

B. Argument

1. First Ground of Rejection (Claims 1-10)

Claims 1-10 are rejected under 35 U.S.C. §103(a) as being unpatentable over *Aucsmith* in view of *Roberts*. Of the rejected claims, Claim 1 is independent. Applicant respectfully submits that independent Claim 1 is patentable over the *Aucsmith* and *Roberts* references and, therefore, Claims 2-10 that depend from independent Claim 1 are also patentable.

Generally, embodiments of the present invention are directed toward a system (10) and method for secure data transmission. For example, in some embodiments of the present invention, a string generator (40) is used to generate character string (54) (at least at page 4, lines 19-21, and figure 1). A hashing engine (42) hashes the character string (54) with a private key (62) to generate a hash key (64) (at least at page 4, lines 20-22, and figure 1). An encryption engine (44) is used to encrypt data (70, 72) using the hash key (64) as an encryption password (at least at page 4, lines 22-24, and figure 1). The encrypted data (72) is transmitted to a recipient along with the character string (54) and an identification key (60) where the identification key (60) identifies the source of the transmitted encrypted data (72) (at least at page 4, lines 7-11 and lines 24-26, and figure 1). The recipient of the encrypted data (72, 112) uses the received identification key (60) to identify the source of the encrypted data (72, 112) and, based on the source of the encrypted data (72, 112), identifies a private key (110) associated with the source of the transmitted data (72, 112) (at least at page 65-112, and figure 1). The recipient of the encrypted data (72, 112) uses a hashing engine (86) to generate a hash key (118) by hashing the received character string (54, 116) with the private key (110) corresponding to the source of the data (72, 112) (at least at page 6, lines 5-16, and figure 1). The recipient uses the generated hash key (118) to decrypt the encrypted data (72, 112). Thus, Claim 1, for example, recites "generating a character string at a sender," "generating a hash key using the character string and a private key," "encrypting the data using the hash key" and "transmitting an identification key associated with the sender, the character string, and the encrypted data from the sender to a recipient."

In the Final Office Action, the Examiner appears to state that *Aucsmith* teaches the limitations of Claim 1 except for generating a hash key using a character string and a private key where the character string is randomly generated (Final Office Action, page 4). Applicant respectfully disagrees. *Aucsmith* appears to disclose that a "[s]ignature generation unit 221 generates a signature of [an] executable program that is a function of all the characters in the file" (*Aucsmith*, column 5, lines 59-61). *Aucsmith* also appears to disclose that "[e]ncryption unit 230 operates to encrypt the executable program by performing an encryption algorithm using the signature . . . as a key" to form an encrypted executable image (*Aucsmith*, column 5, line 65 to column 6, line 1). *Aucsmith* further appears to disclose that "[b]oth the encrypted executable image and the signature are sent as a file to a computer system to be executed." (*Aucsmith*, column 6, lines 6 and 7)(emphasis added). Thus, *Aucsmith* does not appear to disclose or even suggest a "private key" as recited by independent Claim 1. To the contrary, the "key" used to encrypt the executable program of *Aucsmith* is the program itself (e.g., a function of all the characters in the file) and, therefore, is not a "private key." Nor can the composite key of *Aucsmith* be considered the "private key" recited by Claim 1 ("The composite key is associated with specific access rights that are granted to the executable program." (*Aucsmith*, column 6, lines 50-52)). Further, *Aucsmith* does not appear to disclose or even suggest "transmitting an identification key associated with the sender" as recited by independent Claim 1 (emphasis added). Additionally, *Roberts* does not appear to remedy, nor did the Examiner appear to rely on *Roberts* to remedy, at least these deficiencies of *Aucsmith.* Therefore, for at least these reasons, Applicant respectfully submits that Claim 1 is patentable over the *Aucsmith* and *Roberts* references.

In the Final Office Action, the Examiner appears to rely on "an identification mark" of *Aucsmith* that is transmitted in the "executable program" of *Aucsmith* to purportedly correspond to the "identification key" recited by Claim 1 (Final Office Action, page 4 (referring to *Aucsmith*, column 6, lines 39-52)). Applicant respectfully disagrees. The identification mark of *Aucsmith* apparently relied on by the Examiner does not appear to be "associated with the sender" as recited by independent Claim 1. Rather, the identification mark of *Aucsmith* apparently relied on by the Examiner appears to be used to indicate specific access rights that are granted to the executable program (*Aucsmith*, column 6, lines 50-52). Nor does the composite key of *Aucsmith* relate in any manner to "an identification key associated with the sender" as recited by Claim 1 ("The composite key is associated with specific access rights that are granted to the executable program." (*Aucsmith*, column 6, lines 50-52)). Accordingly, for at least these reasons also, Applicant respectfully submits that Claim 1 is patentable over the proposed combination of *Aucsmith* and *Roberts.*

Additionally, the Examiner appears to cite *Roberts* for the purpose of disclosing generating a hash key using the character string and a private key wherein the character string is randomly generated (Final Office Action, pages 4 and 5). The Examiner states that "it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the method of key generation of *Aucsmith* to integrate the concept of generating encryption/decryption keys by hashing a random seed with a private key . . . ." (Final Office Action, page 5). Applicant respectfully disagrees. Applicant respectfully submits that there is no suggestion or motivation to combine the references as proposed by the Examiner. For example, *Aucsmith* discloses that "[s]ignature generation unit 221 generates a signature of the executable program that is a function of all the characters in the file" (*Aucsmith*, column 5, lines 59-61) (emphasis added). *Aucsmith* further discloses that by generating a signature "that is a function of all the characters in the file . . . if the executable program is modified, one would be able to detect the modification by recomputing the cryptographic keyed hash value and comparing the computed value with the original signature" (*Aucsmith*, column 5, lines 59-64) (emphasis added). Accordingly, Applicant respectfully submits that there is no motivation or suggestion to combine reference teachings as suggested by the Examiner at least because to encrypt the executable file of *Aucsmith* as suggested by the Examiner with a randomly generated seed as proposed by the Examiner would prevent one from being able to detect changes to the executable file as taught and desired by *Aucsmith.* In fact, *Aucsmith* teaches away from the proposed combination at least because encrypting the executable file with a randomly generated seed as proposed by the Examiner would preclude at least one apparently important advantage of *Aucsmith* resulting from *Aucsmith's* encrypting the file with "all of the characters of the file." Thus, Applicant respectfully submits that Claim 1 is patentable over the proposed combination of *Aucsmith* and *Roberts*.

Further, as a basis for combining purported reference teachings, the Examiner states that "changing the seed [as purportedly taught by *Roberts*] periodically also changes the key periodically" and that "it would have been obvious . . . to modify the method of key generation of *Aucsmith* to integrate the concept of generating encryption/decryption keys by hashing a random seed with a private key . . . to make it more difficult for eavesdroppers to identify the key" (Final Office Action, page 5). Applicant respectfully disagrees. As discussed above, *Aucsmith* discloses that "a signature of [an] executable program that is a function of all the characters in the file" is generated, and that the generated signature is used to encrypt the executable program file before transmitting the file to another location (*Aucsmith*, column 5, line 59 to column 6, line 7). Thus, for each different file or any change to a file, the signature used to encrypt the file in *Aucsmith* automatically changes because the signature is a function of all characters in the file. Accordingly, there is no motivation or suggestion to combine reference teachings as proposed by the Examiner because the alleged benefit of the proposed combination already appears to be present in the *Aucsmith* reference, namely, a different encryption key. Moreover, the Examiner's statement that "changing the seed [as purportedly taught by *Roberts*] periodically" appears to have no relevance to the *Aucsmith* system at least because the *Aucsmith* system uses the executable program itself (e.g., a function of all the characters in the file) to encrypt the executable program file of *Aucsmith*. Accordingly, for at least this reason also, Applicant respectfully submits that Claim 1 is patentable over the *Aucsmith* and *Roberts* references.

Thus, Applicant respectfully submits that independent Claim 1 is clearly patentable over the *Aucsmith* and *Roberts* references and, therefore, Claims 2-10 that depend therefrom are also patentable.

2. First Ground of Rejection (Claims 11-18)

Claims 11-18 are rejected under 35 U.S.C. §103(a) as being unpatentable over *Aucsmith* in view of *Roberts*. Of the rejected claims, Claim 11 is independent. Applicant respectfully submits that independent Claim 11 is patentable over the *Aucsmith* and *Roberts* references and, therefore, Claims 12-18 that depend from independent Claim 11 are also patentable.

Independent Claim 11 recites "receiving a character string from a sender," "receiving an identification key from the sender," "receiving encrypted data from the sender," "determining a private key associated with the sender using the identification key" and "decrypting the encrypted data using the private key and the character string" (emphasis added). Applicant respectfully submits that the proposed combination of references does not disclose, teach or suggest all limitations of independent Claim 11. For example, as discussed above in connection with independent Claim 1, the Examiner appears to refer to an identification mark of *Aucsmith* as corresponding to the "identification key" recited by Claim 11 (Final Office Action, page 4). Applicant respectfully disagrees. *Aucsmith* recites:

> Identification unit 440 reads an identification mark in the executable program and obtains the identity of a corresponding composite key which is assigned to the identification mark. This composite key is typically the same key used by signature generation unit 221 to generate the keyed hash value of the executable program. In one embodiment of the present invention, identification processor 440 contains a look-up table matching various identification marks with various composite keys. The composite key is associated with specific access rights that are granted to the executable program.

(*Aucsmith*, column 6, lines 41-52)(emphasis added). Accordingly, *Aucsmith* does not "determin[e] a private key associated with the sender using the identification key" or "decrypt[] the encrypted data using the private key [determined using the identification key] and the character string" as recited by independent Claim 11. *Aucsmith*, as stated above, apparently uses an identification mark to identify specific access rights that are granted to the executable program. Additionally, the "identification mark" relied on by the Examiner in *Aucsmith* does not appear to be in any way related to the decryption of the encrypted data in *Aucsmith* ("Decryption unit 430 decrypts the encrypted executable image using the signature component as the decryption key" (*Aucsmith*, column 6, lines 31-36)). Nor can the composite key of *Aucsmith* be considered to be the "private key" recited by Claim 11 ("The composite key is associated with specific access rights that are granted to the executable program." (*Aucsmith*, column 6, lines 50-52)). Further, *Roberts* does not appear to remedy, nor did the Examiner appear to rely on *Roberts* to remedy, at least these deficiencies of *Aucsmith*. Therefore, for at least these reasons, Applicant respectfully submits that independent Claim 11 is patentable over the proposed combination of *Aucsmith* and *Roberts*.

Additionally, the Examiner appears to cite *Roberts* for the purpose of disclosing generating a hash key using the character string and a private key wherein the character string is randomly generated (Final Office Action, pages 6 and 7). The Examiner states that "it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the method of key generation of *Aucsmith* to integrate the concept of generating encryption/decryption keys by hashing a random seed with a private key . . . ." (Final Office Action, page 5). Applicant respectfully disagrees. Applicant respectfully submits that there is no suggestion or motivation to combine the references as proposed by the Examiner. For example, *Aucsmith* discloses that "[s]ignature generation unit 221 generates a signature of the executable program that is a function of all the characters in the file" (*Aucsmith*, column 5, lines 59-61) (emphasis added). *Aucsmith* further discloses that by generating a signature "that is a function of all the characters in the file . . . if the executable program is modified, one would be able to detect the modification by recomputing the cryptographic keyed hash value and comparing the computed value with the original signature" (*Aucsmith*, column 5, lines 59-64). Accordingly, Applicant respectfully submits that there is no motivation or suggestion to combine reference teachings as suggested by the Examiner at least because to encrypt or decrypt the executable file of *Aucsmith* as suggested by the Examiner with a randomly generated seed as proposed by the Examiner would prevent one from being able to detect changes to the executable file as taught and desired by *Aucsmith*. In fact, *Aucsmith* teaches away from the proposed combination at least because encrypting and decrypting the executable file with a randomly generated seed as proposed by the Examiner would preclude at least one apparently important advantage of *Aucsmith* resulting from *Aucsmith's* encrypting the file with "all of the characters of the file." Thus, Applicant respectfully submits that Claim 11 is patentable over the proposed combination of *Aucsmith* and *Roberts*.

Further, as a basis for combining purported reference teachings, the Examiner states that "changing the seed [as purportedly taught by *Roberts*] periodically also changes the key periodically" and that "it would have been obvious . . . to modify the method of key generation of *Aucsmith* to integrate the concept of generating encryption/decryption keys by hashing a random seed with a private key . . . to make it more difficult for eavesdroppers to identify the key" (Final Office Action, page 5). Applicant respectfully disagrees. As discussed above, *Aucsmith* discloses that "a signature of [an] executable program that is a function of all the characters in the file" is generated, and that the generated signature is used to encrypt the executable program file before transmitting the file to another location (*Aucsmith*, column 5, line 59 to column 6, line 7). Thus, for each different file or any change to a file, the signature used to encrypt and decrypt the file in *Aucsmith* automatically changes because the signature is a function of all characters in the file. Moreover, the Examiner's statement that "changing the seed [as purportedly taught by *Roberts*] periodically" appears to have no relevance to the *Aucsmith* system at least because the *Aucsmith* system uses the executable program itself (e.g., a function of all the characters in the file) to encrypt the executable program file of *Aucsmith*. Accordingly, there is no motivation or suggestion to combine reference teachings as proposed by the Examiner because the alleged benefit of the proposed combination already appears to be present in the *Aucsmith* reference, namely, a different encryption/decryption key. Accordingly, for at least these reasons also, Applicant respectfully submits that Claim 11 is patentable over the *Aucsmith* and *Roberts* references.

Thus, Applicant respectfully submits that independent Claim 11 is clearly patentable over the *Aucsmith* and *Roberts* references and, therefore, Claims 12-18 that depend therefrom are also patentable.

3. First Ground of Rejection (Claims 19-26)

Claims 19-26 are rejected under 35 U.S.C. §103(a) as being unpatentable over *Aucsmith* in view of *Roberts*. Of the rejected claims, Claim 19 is independent. Applicant respectfully submits that independent Claim 19 is patentable over the *Aucsmith* and *Roberts* references and, therefore, Claims 20-26 that depend from independent Claim 19 are also patentable.

Independent Claim 19 recites "a hashing engine . . . adapted to generate a hash key using [a] character string and a private key," "an encryption engine . . . adapted to encrypt the data using the hash key" and "wherein [a] processor is adapted to transmit the encrypted data, an identification key related to the private key, and the character string to a recipient" (emphasis added). In the Final Office Action, the Examiner appears to state that *Aucsmith* teaches the limitations of Claim 19 except for generating a hash key using a character string and a private key where the character string is randomly generated (Final Office Action, page 4). Applicant respectfully disagrees. *Aucsmith* appears to disclose that a "[s]ignature generation unit 221 generates a signature of [an] executable program that is a function of all the characters in the file" (*Aucsmith*, column 5, lines 59-61). *Aucsmith* also appears to disclose that "[e]ncryption unit 230 operates to encrypt the executable program by performing an encryption algorithm using the signature . . . as a key" to form an encrypted executable image (*Aucsmith*, column 5, line 65 to column 6, line 1). *Aucsmith* further appears to disclose that "[b]oth the encrypted executable image and the signature are sent as a file to a computer system to be executed." (*Aucsmith*, column 6, lines 6 and 7)(emphasis added). Thus, *Aucsmith* does not appear to disclose or even suggest a "private key" as recited by independent Claim 19. To the contrary, the "key" used to encrypt the executable program of *Aucsmith* is the program itself (e.g., a function of all the characters in the file) and, therefore, is not a "private key." Nor can the composite key of *Aucsmith* be considered the "private key" recited by Claim 19 ("The composite key is associated with specific access rights that are granted to the executable program." (*Aucsmith*, column 6, lines 50-52)). Further, *Aucsmith* does not appear to disclose or even suggest "transmit[ting] the encrypted data, an identification key related to [a] private key, and the character string to a recipient" as recited by independent Claim 19 (emphasis added). Additionally, *Roberts* does not appear to remedy, nor did the Examiner appear to rely on *Roberts* to remedy, at least theses deficiencies of *Aucsmith*. Therefore, for at least these reasons, Applicant respectfully submits that Claim 19 is patentable over the *Aucsmith* and *Roberts* references.

In the Final Office Action, the Examiner appears to rely on "an identification mark" of *Aucsmith* that is transmitted in the "executable program" of *Aucsmith* to purportedly correspond to the "identification key" recited by Claim 19 (Final Office Action, page 4 (referring to *Aucsmith*, column 6, lines 39-52)). Applicant respectfully disagrees. The identification mark of *Aucsmith* apparently relied on by the Examiner appears to be used in the *Aucsmith* reference to indicate specific access rights that are granted to the executable program (*Aucsmith*, column 6, lines 50-52). Accordingly, for at least this reason also, Applicant respectfully submits that Claim 19 is patentable over the proposed combination of *Aucsmith* and *Roberts*.

Additionally, the Examiner appears to cite *Roberts* for the purpose of disclosing generating a hash key using the character string and a private key wherein the character string is randomly generated (Final Office Action, pages 4 and 5). The Examiner states that "it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the method of key generation of *Aucsmith* to integrate the concept of generating encryption/decryption keys by hashing a random seed with a private key . . . ." (Final Office Action, page 5). Applicant respectfully disagrees. Applicant respectfully submits that there is no suggestion or motivation to combine the references as proposed by the Examiner. For example, *Aucsmith* discloses that "[s]ignature generation unit 221 generates a signature of the executable program <u>that is a function of all the characters in the file</u>" (*Aucsmith*, column 5, lines 59-61) (emphasis added). *Aucsmith* further discloses that by generating a signature "that is a function of <u>all</u> the characters in the file . . . if the executable program is modified, one would be able to detect the modification by recomputing the cryptographic keyed hash value and comparing the computed value with the original signature" (*Aucsmith*, column 5, lines 59-64). Accordingly, Applicant respectfully submits that there is no motivation or suggestion to combine reference teachings as suggested by the Examiner at least because to encrypt the executable file of *Aucsmith* as suggested by the Examiner with a randomly generated seed as proposed by the Examiner would prevent one from being able to detect changes to the executable file as taught and desired by *Aucsmith*. In fact, *Aucsmith* teaches away from the proposed combination at least because encrypting the executable file with a randomly generated seed as proposed by the Examiner would preclude at least one apparently important advantage of *Aucsmith* resulting from *Aucsmith's* encrypting the file with "all of the characters of the file." Thus, Applicant respectfully submits that Claim 19 is patentable over the proposed combination of *Aucsmith* and *Roberts*.

Further, as a basis for combining purported reference teachings, the Examiner states that "changing the seed [as purportedly taught by *Roberts*] periodically also changes the key periodically" and that "it would have been obvious . . . to modify the method of key generation of *Aucsmith* to integrate the concept of generating encryption/decryption keys by hashing a random seed with a private key . . . to make it more difficult for eavesdroppers to identify the key" (Final Office Action, page 5). Applicant respectfully disagrees. As discussed above, *Aucsmith* discloses that "a signature of [an] executable program that is a function of all the characters in the file" is generated, and that the generated signature is used to encrypt the executable program file before transmitting the file to another location (*Aucsmith*, column 5, line 59 to column 6, line 7). Thus, for each different file or any change to a file, the signature used to encrypt the file in *Aucsmith* automatically changes <u>because the signature is a function of all characters in the file</u>. Moreover, the Examiner's statement that "changing the seed [as purportedly taught by *Roberts*] periodically" appears to have no relevance to the *Aucsmith* system at least because the *Aucsmith* system uses the executable program itself (e.g., a function of all the characters in the file) to encrypt the executable program file of *Aucsmith*. Accordingly, there is no motivation or suggestion to combine reference teachings as proposed by the Examiner because the alleged benefit of the proposed combination already appears to be present in the *Aucsmith* reference, namely, a different encryption key. Accordingly, for at least these reasons also, Applicant respectfully submits that Claim 19 is patentable over the *Aucsmith* and *Roberts* references.

Thus, Applicant respectfully submits that independent Claim 19 is clearly patentable over the *Aucsmith* and *Roberts* references and, therefore, Claims 20-26 that depend therefrom are also patentable.

4. First Ground of Rejection (Claims 27-34)

Claims 27-34 are rejected under 35 U.S.C. §103(a) as being unpatentable over *Aucsmith* in view of *Roberts*. Of the rejected claims, Claim 27 is independent. Applicant respectfully submits that independent Claim 27 is patentable over the *Aucsmith* and *Roberts* references and, therefore, Claims 28-34 that depend from independent Claim 27 are also patentable.

Independent Claim 27 recites "a processor adapted to receive encrypted data, an identification key, and a character string from a sender," "a relational database stored in [a] memory . . . relating the identification key to a private key" and "a decryption engine . . . adapted to decrypt the encrypted data using the character string and the private key" (emphasis added). Applicant respectfully submits that the proposed combination of references does not disclose, teach or suggest all limitations of independent Claim 27. For example, as discussed above in connection with independent Claim 1, the Examiner appears to refer to an identification mark of *Aucsmith* as corresponding to the "identification key" recited by Claim 27 (Final Office Action, page 4). Applicant respectfully disagrees. *Aucsmith* recites:

> Identification unit 440 reads an identification mark in the executable program and obtains the identity of a corresponding composite key which is assigned to the identification mark. This composite key is typically the same key used by signature generation unit 221 to generate the keyed hash value of the executable program. In one embodiment of the present invention, identification processor 440 contains a look-up table matching various identification marks with various composite keys. The composite key is associated with specific access rights that are granted to the executable program.

(*Aucsmith*, column 6, lines 41-52)(emphasis added). Accordingly, *Aucsmith* does not "receive encrypted data, an identification key, and a character string from a sender," or "decrypt the encrypted data using the character string and the private key" as recited by independent Claim 27. *Aucsmith*, as stated above, apparently uses an identification mark to identify specific access rights that are granted to the executable program. Additionally, the "identification mark" relied on by the Examiner in *Aucsmith* does not appear to be in any way related to the decryption of the encrypted data in *Aucsmith* ("Decryption unit 430 decrypts the encrypted executable image using the signature component as the decryption key" (*Aucsmith*, column 6, lines 31-36)). Further, *Roberts* does not appear to remedy, nor did the Examiner appear to rely on *Roberts* to remedy, at least these deficiencies of *Aucsmith*. Therefore, for at least these reasons, Applicant respectfully submits that independent Claim 27 is patentable over the proposed combination of *Aucsmith* and *Roberts*.

Additionally, the Examiner appears to cite *Roberts* for the purpose of disclosing generating a hash key using the character string and a private key wherein the character string is randomly generated (Final Office Action, pages 6 and 7). The Examiner states that "it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the method of key generation of *Aucsmith* to integrate the concept of generating encryption/decryption keys by hashing a random seed with a private key . . . ." (Final Office Action, page 5). Applicant respectfully disagrees. Applicant respectfully submits that there is no suggestion or motivation to combine the references as proposed by the Examiner. For example, *Aucsmith* discloses that "[s]ignature generation unit 221 generates a signature of the executable program that is a function of all the characters in the file" (*Aucsmith*, column 5, lines 59-61) (emphasis added). *Aucsmith* further discloses that by generating a signature "that is a function of all the characters in the file . . . if the executable program is modified, one would be able to detect the modification by recomputing the cryptographic keyed hash value and comparing the computed value with the original signature" (*Aucsmith*, column 5, lines 59-64). Accordingly, Applicant respectfully submits that there is no motivation or suggestion to combine reference teachings as suggested by the Examiner at least because to encrypt or decrypt the executable file of *Aucsmith* as suggested by the Examiner with a randomly generated seed as proposed by the Examiner would prevent one from being able to detect changes to the executable file as taught and desired by *Aucsmith.* In fact, *Aucsmith* teaches away from the proposed combination at least because encrypting and decrypting the executable file with a randomly generated seed as proposed by the Examiner would preclude at least one apparently important advantage of *Aucsmith* resulting from *Aucsmith's* encrypting the file with "all of the characters of the file." Thus, Applicant respectfully submits that Claim 27 is patentable over the proposed combination of *Aucsmith* and *Roberts*.

Further, as a basis for combining purported reference teachings, the Examiner states that "changing the seed [as purportedly taught by *Roberts*] periodically also changes the key periodically" and that "it would have been obvious . . . to modify the method of key generation of *Aucsmith* to integrate the concept of generating encryption/decryption keys by hashing a random seed with a private key . . . to make it more difficult for eavesdroppers to identify the key" (Final Office Action, page 5). Applicant respectfully disagrees. As discussed above, *Aucsmith* discloses that "a signature of [an] executable program that is a function of all the characters in the file" is generated, and that the generated signature is used to encrypt the executable program file before transmitting the file to another location (*Aucsmith*, column 5, line 59 to column 6, line 7). Thus, for each different file or any change to a file, the signature used to encrypt and decrypt the file in *Aucsmith* automatically changes because the signature is a function of all characters in the file. Accordingly, there is no motivation or suggestion to combine reference teachings as proposed by the Examiner because the alleged benefit of the proposed combination already appears to be present in the *Aucsmith* reference, namely, a different encryption/decryption key. Accordingly, for at least this reason also, Applicant respectfully submits that Claim 27 is patentable over the *Aucsmith* and *Roberts* references.

Thus, Applicant respectfully submits that independent Claim 27 is clearly patentable over the *Aucsmith* and *Roberts* references and, therefore, Claims 28-34 that depend therefrom are also patentable.

CONCLUSION

Applicant has demonstrated that the present invention as claimed is clearly distinguishable over the art cited of record. Therefore, Applicant respectfully requests the Board of Patent Appeals and Interferences to reverse the final rejection of the Examiner and instruct the Examiner to issue a notice of allowance of all claims.

The Commissioner is authorized to charge the statutory fee of $500.00 to Deposit Account No. 08-2025 of Hewlett-Packard Company. Although no other fee is believed due, the Commissioner is hereby authorized to charge any fees or credit any overpayments to Deposit Account No. 08-2025 of Hewlett-Packard Company.

Respectfully submitted,

James L Baudino

James L. Baudino
Registration No. 43,486

Date: April 27, 2006

Correspondence To:

L. Joy Griebenow
Hewlett-Packard Company
Intellectual Property Administration
P.O. Box 272400
Fort Collins, Colorado 80527-2400
Tel. (970) 898-3884

CLAIMS APPENDIX

1. A method for secure data transmission, comprising:

generating a character string at a sender;

generating a hash key using the character string and a private key;

encrypting the data using the hash key; and

transmitting an identification key associated with the sender, the character string, and the encrypted data from the sender to a recipient.

2. The method of Claim 1, wherein generating the hash key comprises hashing the character string with the private key.

3. The method of Claim 1, further comprising:

generating a signature using the hash key and the data; and

transmitting the signature from the sender to the recipient.

4. The method of Claim 1, wherein generating a character string comprises randomly generating the character string.

5. The method of Claim 1, further comprising:

determining the private key at the recipient using the identification key; and

decrypting the encrypted data at the recipient using the private key and the character string.

6. The method of Claim 5, wherein determining the private key comprises accessing a relational database associating the identification key to the private key.

7. The method of Claim 1, further comprising:

determining the private key at the recipient using the identification key;

determining the hash key at the recipient using the private key and the character string; and

decrypting the encrypted data using the hash key.

8. The method of Claim 7, wherein determining the hash key comprises hashing the private key with the character string.

9. The method of Claim 1, further comprising:

generating a first signature by the sender using the hash key and the data; and

transmitting the first signature to the recipient, the recipient adapted to determine the hash key for decrypting the data and compare the first signature to a second signature generated by the recipient using the hash key and the decrypted data.

10. The method of Claim 1, further comprising:

generating a signature using the hash key and the data;

transmitting the signature to the recipient;

determining the private key at the recipient using the identification key;

determining the hash key at the recipient using the private key and the character string;

decrypting the encrypted data at the recipient using the hash key; and

verifying the signature at the recipient using the hash key and the decrypted data.

11. A method for secure data transmission, comprising:

receiving a character string from a sender;

receiving an identification key from the sender;

receiving encrypted data from the sender;

determining a private key associated with the sender using the identification key; and

decrypting the encrypted data using the private key and the character string.

12. The method of Claim 11, further comprising determining a hash key using the character string and the private key, and wherein decrypting the encrypted data comprises decrypting the encrypted data using the hash key.

13. The method of Claim 11, wherein determining the private key comprises accessing a relational database associating the identification key to the private key.

14. The method of Claim 11, wherein receiving the character string comprises receiving a randomly generated character string.

15. The method of Claim 11, further comprising hashing the character string with the private key to generate a hash key, and wherein decrypting the encrypted data comprises decrypting the encrypted data using the hash key.

16. The method of Claim 11, further comprising:

receiving a signature from the sender; and

verifying the signature using the decrypted data, the private key, and the character string.

17. The method of Claim 11, further comprising:

receiving a signature from the sender;

determining a hash key using the private key and the character string; and

verifying the signature using the decrypted data and the hash key.

18. The method of Claim 11, further comprising:

receiving a first signature from the sender;

determining a hash key using the private key and the character string;

generating a second signature using the hash key and the decrypted data; and

comparing the first signature to the second signature.

19. A system for secure data transmission, comprising:

a processor;

a memory coupled to the processor;

a string generator stored in the memory and executable by the processor, the string generator adapted to generate a character string;

a hashing engine stored in the memory and executable by the processor, the hashing engine adapted to generate a hash key using the character string and a private key; and an encryption engine stored in the memory and executable by the processor, the encryption engine adapted to encrypt the data using the hash key; and

wherein the processor is adapted to transmit the encrypted data, an identification key related to the private key, and the character string to a recipient.

20. The system of Claim 19, further comprising a signature engine stored in the memory and executable by the processor, the signature engine adapted to generate a signature using the hash key and the data, the processor further adapted to transmit the signature to the recipient.

21. The system of Claim 20, wherein the recipient is adapted to decrypt the encrypted data and verify the signature using the decrypted data.

22. The system of Claim 19, wherein the hashing engine is adapted to hash the character string with the private key to generate the hash key.

23. The system of Claim 19, wherein the string generator is adapted to randomly generate the character string.

24. The system of Claim 19, wherein the recipient is adapted to decrypt the encrypted data using the identification key and the character string.

25. The system of Claim 19, wherein the recipient is adapted to determine the hash key using the identification key and the character string and decrypt the encrypted data using the hash key.

26. The system of Claim 19, wherein the recipient is adapted to access a relational database associating the identification key with the private key and decrypt the encrypted data using the private key and the character string.

27. A system for secure data transmission, comprising:

a processor adapted to receive encrypted data, an identification key, and a character string from a sender;

a memory coupled to the processor;

a relational database stored in the memory and accessible by the processor, the relational database relating the identification key to a private key; and

a decryption engine stored in the memory and executable by the processor, the decryption engine adapted to decrypt the encrypted data using the character string and the private key.

28. The system of Claim 27, further comprising a hashing engine stored in the memory and executable by the processor, the hashing engine adapted to generate a hash key using the private key and the character string, the decryption engine adapted to decrypt the encrypted data using the hash key.

29. The system of Claim 27, further comprising a signature engine stored in the memory and executable by the processor, the signature engine adapted to verify a signature received from the sender using the private key and the character string.

30. The system of Claim 27, further comprising:

a hashing engine stored in the memory and executable by the processor, the hashing engine adapted to generate a hash key using the private key and the character string; and

a signature engine stored in the memory and executable by the processor, the signature engine adapted to verify a signature received from the sender using the hash key and the decrypted data.

31. The system of Claim 27, further comprising a hashing engine stored in the memory and executable by the processor, the hashing engine adapted to hash the character string with the private key to generate a hash key, the decryption engine adapted to decrypt the encrypted data using the hash key.

32. The system of Claim 27, further comprising a string generator stored in the memory and executable by the processor, the string generator adapted to generate a

character string, and wherein the decryption engine is further adapted to encrypt-data for transmitting to the sender using the character string and the private key.

33. The system of Claim 32, further comprising:

a string generator stored in the memory and executable by the processor, the string generator adapted to generate a character string; and

a hashing engine stored in the memory and executable by the processor, the hashing engine adapted to hash the character string with the private key to generate a hash key, and wherein the decryption engine is further adapted to encrypt data for transmitting to the sender using the hash key.

34. The system of Claim 32, further comprising a signature engine stored in the memory and executable by the processor, the signature engine adapted to generate a first signature using the decrypted data and compare the first signature to a second signature received from the sender.

EVIDENCE APPENDIX

None

RELATED PROCEEDINGS APPENDIX

None